The Future of Engineering Education
2024 Annual Conference & Exposition
Oregon Convention Center
Portland, OR . June 23 - 26, 2024
ASEE
Paper ID #42811

Analysis of Chemical Engineering Curricula Using Graph Theory

Blake Lopez, University of Wisconsin, Madison

University of Wisconsin-Madison Chemical Engineering PhD Candidate

Work in Progress: Analysis of Chemical Engineering Curricula using Graph Theory

Blake Lopez*† and Victor M. Zavala†

†Department of Chemical and Biological Engineering
University of Wisconsin - Madison, 1415 Engineering Dr, Madison, WI 53706, USA

Abstract

Chemical engineering is a complex interconnected major. Just as chemical engineers have broken complex processes into unit operations, the chemical engineering curriculum has been broken up into courses. The organization of these courses varies amongst institutions and are based on years of prior teaching and research. Despite this, there have been calls to reevaluate the curriculum from both industry and academia. We propose a graph-based representation of curricula in which topics are represented by nodes and topic dependencies are represented by directed edges forming a directed acyclic graph. This enables using graph theory measures and tools to provide formal ways of evaluating a curriculum. Additionally, the abstraction is readily understandable meaning conversations between instructors regarding the curriculum can occur within a department and even across institutions. This abstraction is explained with a simplified curriculum and applied to the undergraduate chemical engineering curriculum of a subject university. Highly and lowly connected topics are identified and approaches for grouping the topics into modules are discussed.

Keywords: Modeling; Curriculum; Graph Theory; Modularity

*Corresponding Author: brlopez@wisc.edu

1 Introduction

The early days of chemical engineering focused on tools such as unit operations, material and energy balances, process control, transport phenomena, and more to develop petrochemicals [1]. Since then, these tools have been increasingly applied to more applications, and chemical engineers can be found in diverse industries and roles. Climate change has also placed pressure on industries to decrease the environmental impact of their products and processes. However, there are proven difficulties in embedding sustainability in engineering curricula, and industry has identified ability shortages in graduating chemical engineers [2]. For these reasons and more, there have been calls to evaluate the chemical engineering curriculum [3, 4] and some departments have even elected to not seek accreditation to allow students more personalized paths towards different careers [5].

One critical aspect of a chemical engineering program is course(s) on process/plant design. Process design is often treated as the culmination of the undergraduate chemical engineering curriculum. This course draws on knowledge from previous chemical engineering courses including transport, balances, controls, and more. Students are often reminded of what they learned in previous courses and are first required to pull knowledge from multiple previous courses. This means students are likely to see the interconnectivity of the curricula for the first time at the very end of their education. Additionally, process design tends to focus on the design of a single process typically within the chemicals or petrochemical sector. This means that students interested in other fields such as biological systems or semiconductors may be less motivated in this course. For these reasons, there is discourse on implementing systems thinking and design throughout the curriculum instead of tying the curriculum together at the very end of a student's learning experience.

All these issues mean the curriculum needs to be evaluated to improve outcomes. There are some tools to analyze curricula, such as curriculum prerequisite maps to identify bridge and source-hub courses, tree-structured topic modeling, connection coefficient determination, and concept maps [6, 7, 8, 9, 10, 11]. Concept maps have been applied to chemical engineering courses and curricula to enhance student learning and explore the ontology of topics covered [12, 13]. Other work has applied statistical tools to study how chemical engineering curricula have changed focus and analyzed different sequencing of transport courses at institutions [14].

We propose a graph-based representation in which topics are represented by nodes and dependencies between topics are represented by directed edges. Unlike in concept maps, the directed edges do not have words attached to them. This enables the use of graph tools to guide curriculum analysis with numerical measures to formalize discussion in improving a curriculum.

2 Graph Abstraction

To explain graph theory applied to a curriculum, a simplified Curriculum A will be used. Curriculum A is composed of six courses which are named ABC 101-106. The courses depend on one another through prerequisites. For example, ABC 101 is a prerequisite to complete ABC 103. The full list of prerequisites and the credits for each course can be found in Table 1.

Table 1: Curriculum A courses, credits, and prerequisites.

Course	Credits	Prerequisite(s)
ABC 101	3	None
ABC 102	4	None
ABC 103	4	ABC 101
ABC 104	2	ABC 102
ABC 105	2	ABC 102, ABC 103
ABC 106	3	ABC 105

This table looks like one a student would find in their handbook or when registering for the courses for the term. This table can be represented with a graph abstraction in which courses are represented by nodes and directional edges connecting nodes represent dependencies as seen in Figure 1. The edges are directional to represent the sequence of the courses and follows the thinking of 'I have to take this course before I take that course'. With the nature of course progression, there are no cycles. For example, ABC 103 would not and cannot be a prerequisite for ABC 101 because then there would be no way to take either of them. This makes the graph abstraction both directional and acyclic. Within this representation, the number of credits for each course is reflected by the node size.

The course level graph abstraction can allow for the scheduling of one's term and visualizing how courses interact with one another. However, these dependencies rely on the accuracy of the prerequisites which could have been established years prior and the courses and, more broadly, the curriculum could have changed since. For example, students may be encouraged by their adviser to take ABC 103 and ABC 104 in the same term. From the graph abstraction, this need is not clear perhaps because ABC 103 has topics that are required for ABC 106 despite not showing up in the perquisites. This means looking more closely into the courses and their interconnectivity is required to fully understand the curriculum.

To look more closely at the curriculum, we can develop topic level graphs for each course within Curriculum A. This could be generated from the schedule of topics in a syllabus, recommended textbook readings, and more. The selection of topics and their connectivity takes some expertise and refinement, but the generation of the abstraction can provide invaluable insights. The graph abstraction for ABC 101 is presented in Figure 2.

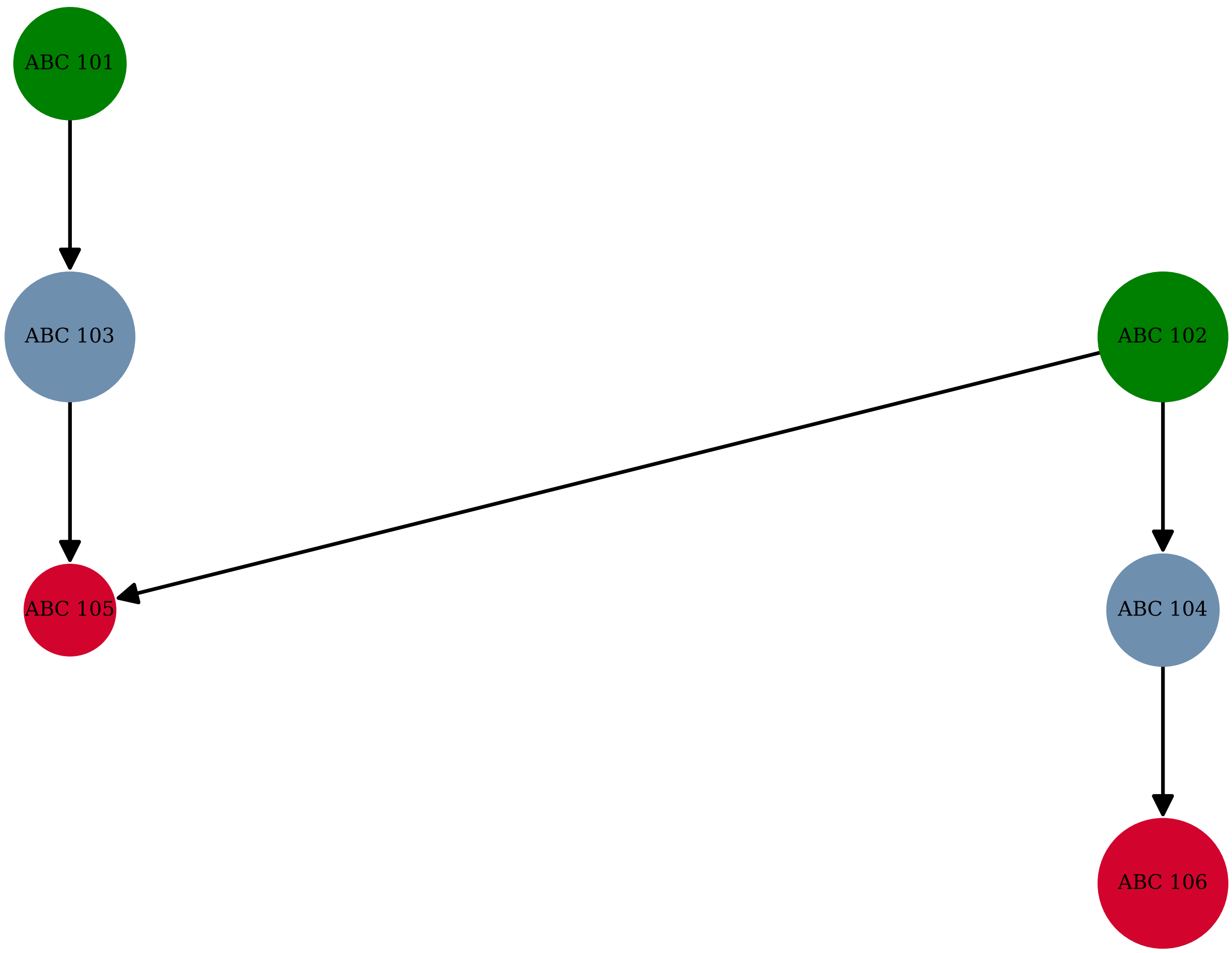

Figure 1: Course level graph abstraction of Curriculum A in which courses are represented as nodes and directed edges represent course dependencies. Node size represents the credit hours for the course. Courses without prerequisites are green, terminal courses are red, and courses with both prerequisites and subsequent dependencies are blue.

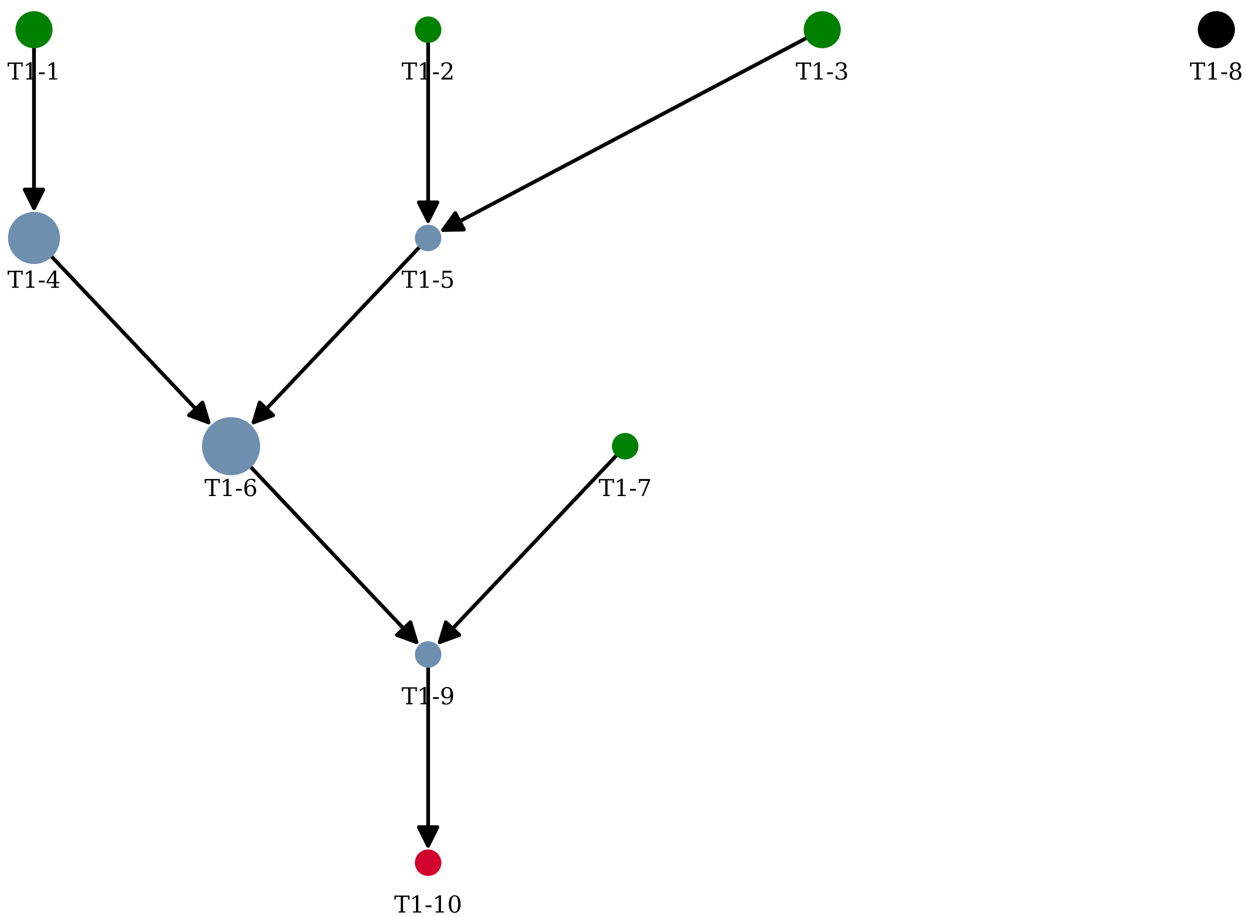

Figure 2: Graph abstraction of ABC 101 in which nodes are the individual topics that make up the course and directional edges represent the dependencies between courses. Node size reflects the relative time the topic is covered in the course in which the sum of all of the node sizes is the credit hours for the course.

In this topic level graph abstraction of ABC 101, topics are represented by nodes and the size of the nodes represent the relative time spent covering the topic in terms of the credit hours for the course. This means the sum of all the node sizes is the number of credit hours for the course. Topic connectivity is represented by the directional edges. A topic without any preceding topics, such as T1-1, means it can be taught without any other prior knowledge. These can be thought of as introductory topics like how one would think about an introductory course. A topic with preceding topics, such as T1-6, means it requires the preceding connected topics (T1-4 and T1-5) to be taught. This follows the logic of ”I need to know numbers and counting to learn addition.” These dependencies can also be used to help plan the course by noticing topic T1-10 requires all other topics besides T1-8. Topics not connected to any others, such as T1-8, means it can be taught at any time during the course because it does not build on previous topics, nor is it required to understand other topics.

While the graph abstraction of ABC 101 provides a lot of valuable insights, it is not the complete picture. ABC 101 is a prerequisite for ABC 103, but this is not yet indicated. Expanding the topic level abstraction to include all courses will provide the complete picture of the curriculum and enable understanding of the interconnectivity of the topics and courses. This is accomplished by constructing a graph abstraction of each course and then connecting them where appropriate. This results in the topic level graph abstraction of Curriculum A in Figure 3.

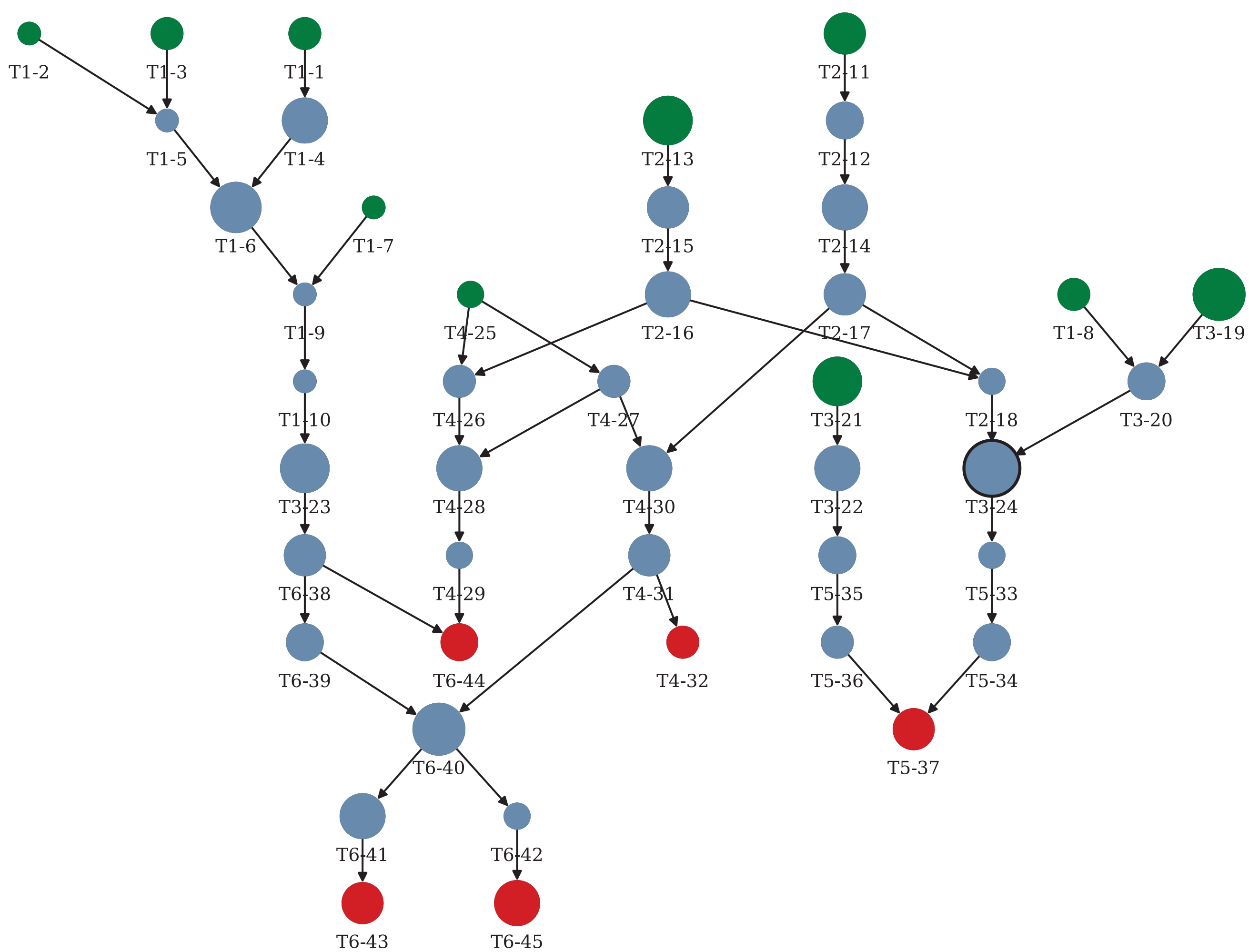

Figure 3: Topic level graph abstraction Curriculum A in which topics are represented by nodes and topic dependencies are represented by directed edges forming a directed acyclic graph. Node size reflects the relative resources required for the topic in terms of the credit hours for the course.

Topics can be introduced in one course and later reviewed in another. Reviewing a topic and then building off it (connecting to another topic) is different than just building off one topic to another because of the time spent on the review. To reflect this, if a topic is introduced and later reviewed, then the size of the topic includes the first time it was taught and any other review times. To distinguish between the two, a border will be added to the node. For example, T3-24 is introduced in ABC 103, but is later reviewed in ABC 105. As such the inner circle represents the resources first spent learning T3-34 and the outside circle represents the resources spent reviewing.

The topic level graph of Curriculum A can be very useful on its own to understand the scope of the curriculum and discussion between instructors. However, the analysis can be improved by using tools from graph theory to provide formalized measures to discuss ways of changing the curriculum. Measures like the degree of node (the number of edges connected to a node), determining paths (getting from one node to another without any repeat nodes), the number of components (subsets of nodes that do not have edges to other nodes), connected vs. strongly connected graphs, and more can be considered. For this analysis, grouping of nodes through community detection and modularity and the degree of a node will be critical for analysis.

For grouping of nodes, a common approach is known as community detection which seeks to maximize "modularity" using optimization. The "modularity" is a function of the interconnectivity of communities that make up the graph. These algorithms can be useful to identify which topics form areas of dense interconnectivity, but it does not consider the size of the nodes. This means it may result in communities of varying sizes. For example, applying the community detection algorithm known as greedy modularity maximization to Curriculum A results in the communities in Figure 4. Here the number of communities was restricted to be equal to the number of original courses (six). It is important to note that the topics forming a community are not always from the same course. The community sizes are 5.8, 3.4, 3.2, 2.7, 2.65, and 2.25 credit hours which when turned into courses could lead to an unbalanced course load.

To improve upon the community detection, a different approach to modularity will be considered. This approach described in [15] can restrict the sizes of the module in addition to the number of modules. This approach maximizes a different measure of modularity with constraints to set the number of modules and the size of modules within a given size. This is cast as a binary quadratic optimization program. Additionally, we can use the framework to provide a definition of a modular curriculum. A curriculum will be considered modular if: (a) the topics in a module form a cluster of dense dependency, (b) connectivity between modules is sparse, and (c) the modules are within a set resource use (credit hours). These two approaches enable evaluating the modularity of the curriculum by determining modules that could be used to guide course organization and overall curriculum planning.

Restricting the module size between 2.9 and 4.1 credit hours, the new topic grouping is presented in Figure 5. This results in community sizes of 3.9, 3.8, 3.4, 3.1, 2.9, and 2.9 credit hours. This approach leads to more balanced course load and modules of more implementable size.

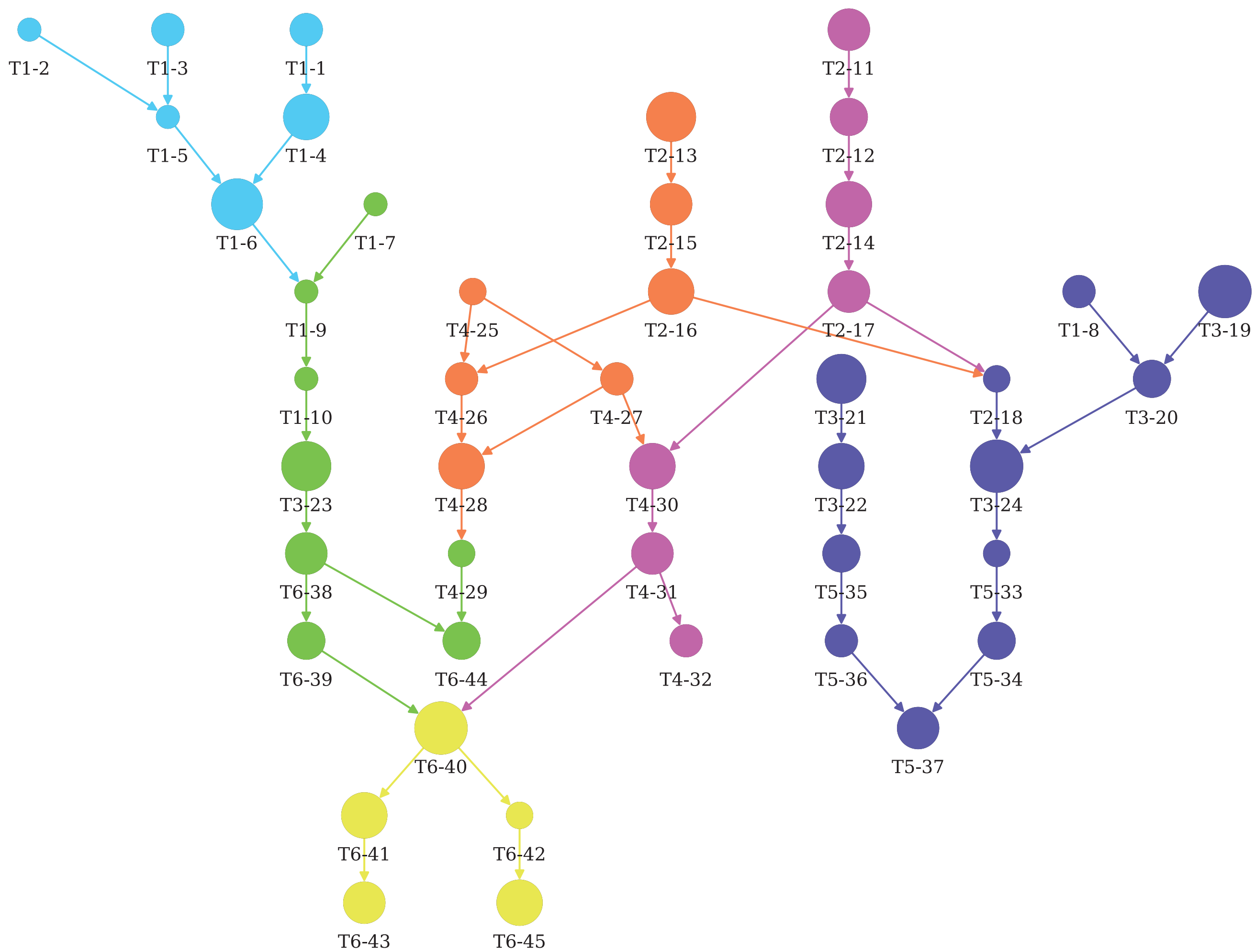

Figure 4: Topic level graph abstraction Curriculum A grouped into six communities (equal to the number of original courses) using greedy community detection where each color represents a different community

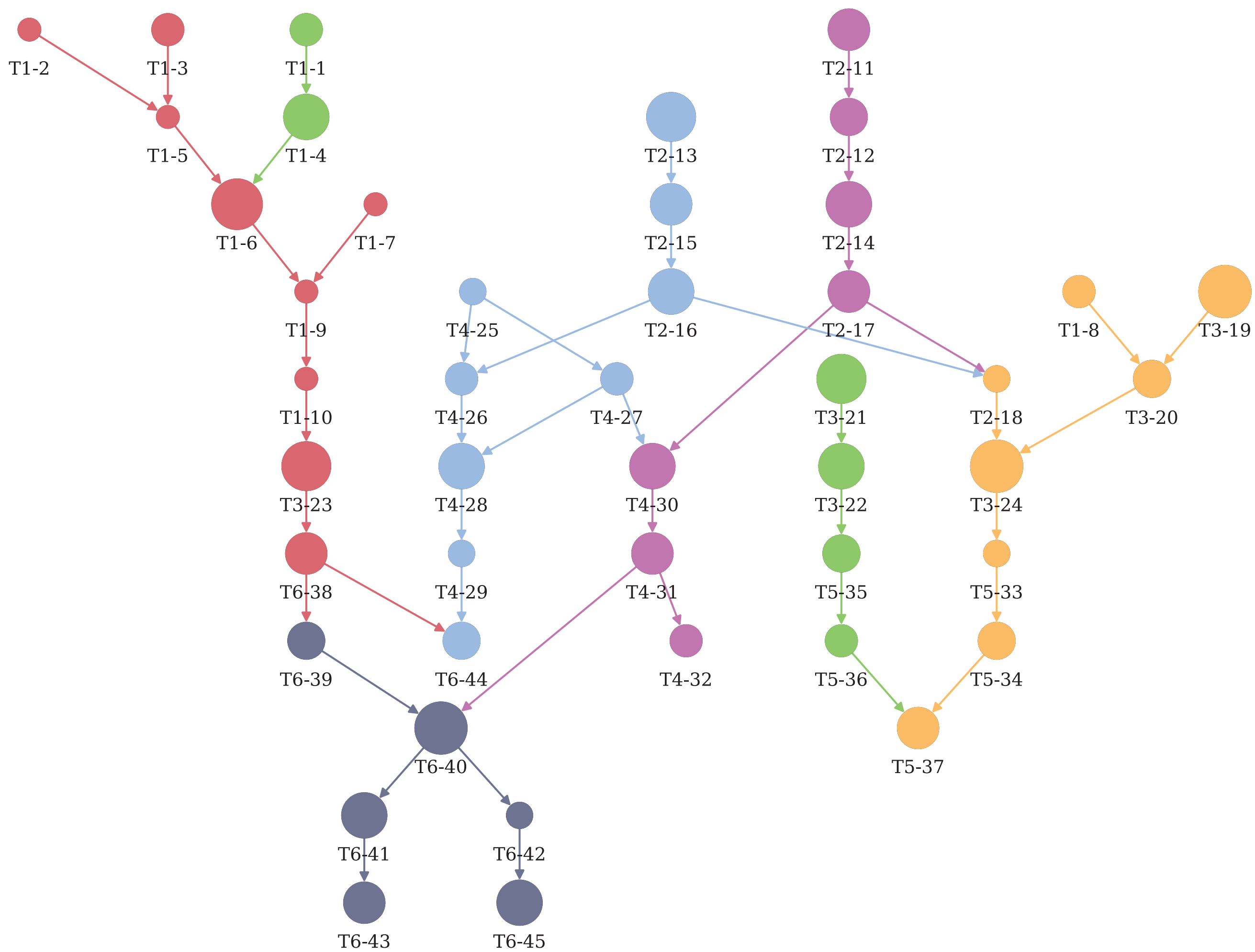

Figure 5: Topic level graph abstraction Curriculum A grouped into six modules (equal to the number of original courses) when restricting module sizes between 2.9 and 4.1 credits.

3 Case Study

3.1 Building Curriculum Graph

To demonstrate some of the capabilities and insights that can be gained from this graph abstraction, this analysis will be applied to the undergraduate chemical engineering curriculum at University of Wisconsin-Madison which serves >400 undergrad and >100 graduate students. This curriculum is composed of 19 credit hours of math, 10 of physics, 20 of chemistry, 6 of life sciences, 6 of communication skills, 16 of liberal studies, 6 of professional depth, 40 of core chemical engineering courses, and 9 chemical engineering electives for a total of 132 semester credit hours. The focus of this analysis will be on the core chemical engineering courses listed in Table 2 along with their already defined prerequisites. Most of these courses are similar to courses required by other undergraduate chemical engineering programs.

Table 2: Core chemical engineering courses at the subject university. Term taken is the term a student would normally take the course where 1F means fall of year one and 4S means spring of year four.

Course	Name	Term	Credits	Prerequisite(s)
CBE 150	Intro Chemical Engineering	1F	1	None
CBE 250	Process Synthesis	2F	3	None
CBE 255	Intro to Chemical Process Modeling	2S	3	CBE 250 (co)
CBE 310	Chemical Process Thermodynamics	2S	3	CBE 250 (co)
CBE 311	Thermodynamics of Mixtures	3F	3	CBE 310, 255
CBE 320	Intro Transport Phenomena	3F	4	None
CBE 324	Transport Phenomena Lab	3S	3	CBE 310, 320
CBE 326	Momentum and Heat Transfer	3S	3	CBE 320 (co)
CBE 426	Mass Transfer Operations	4F	3	CBE 311, CBE 320
CBE 430	Chem Kinetics and Reactor Design	4F	3	CBE 311, CBE 320
CBE 450	Process Design	4S	3	CBE 326, 430, 426
CBE 470	Process Dynamics and Control	4S	3	CBE 326, 430 (co)
CBE 424	Operations Process Lab	4Su	5	CBE 324, 326, 426, 430

A graph abstraction of the core chemical engineering courses is presented in Figure 6. One interesting note of the curriculum is CBE 250 can be taken concurrent with CBE 255 and CBE 255 can be taken concurrent with CBE 310, but CBE 250 is a prerequisite for CBE 310. Since this graph abstraction is made to model the flow of topics, CBE 250 will be treated as a prerequisite CBE 255 and CBE 255 will be treated as a prerequisite for CBE 310. A similar scenario exists for CBE 320 and CBE 324 and CBE 430 and CBE 470. From this course level abstraction, one can easily see just how many courses depend on CBE 320 based on connectivity.

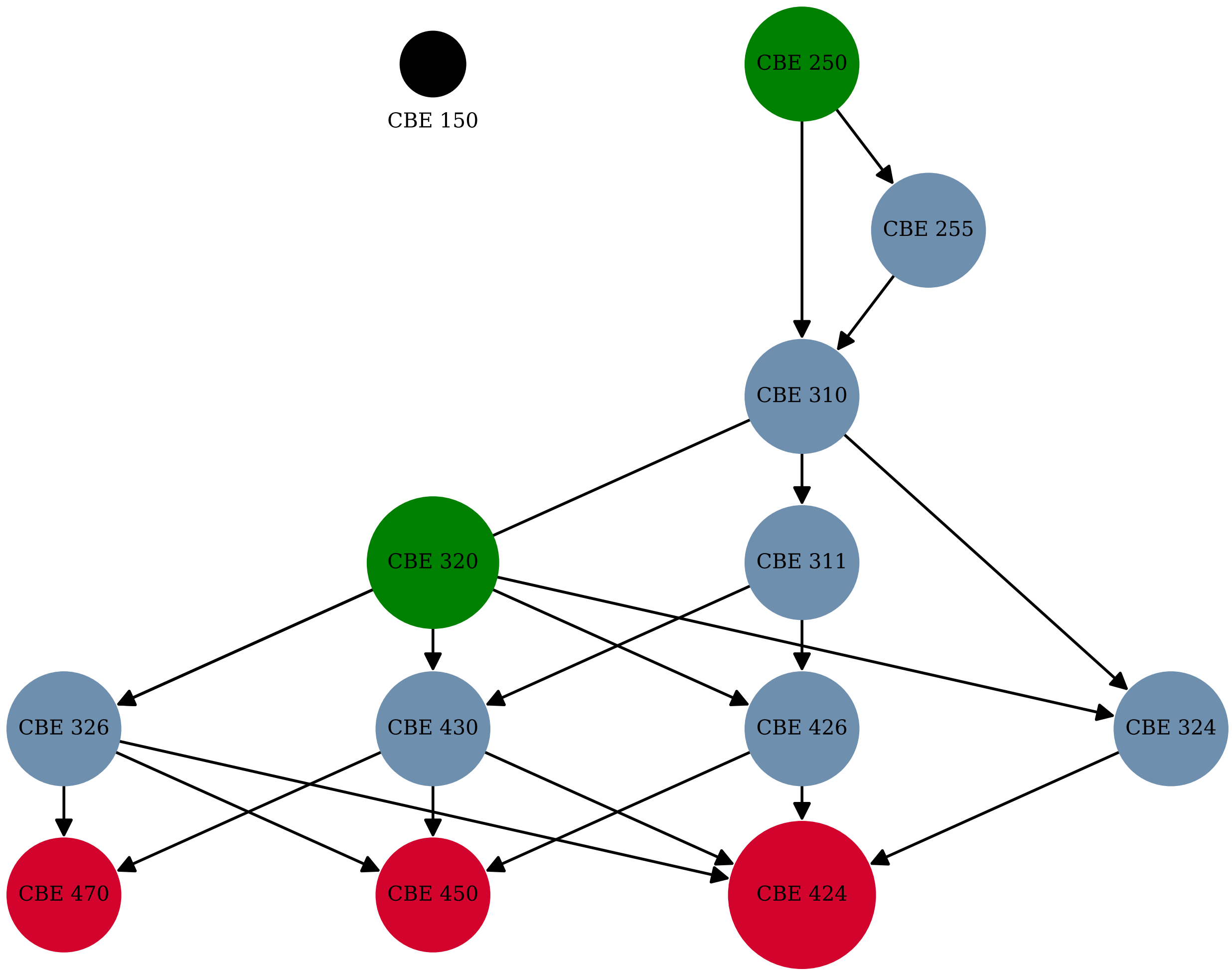

Figure 6: Course level graph abstraction of the core chemical engineering courses.

To construct the topic level graph abstraction of the curriculum, the syllabuses, course schedules, course readings, and lab documents were collected from recent occurrences of the courses. Then, an iterative approach of defining topics and topic dependency was completed to form the graph abstraction for each course. For example, the graph for CBE 320: Introductory Transport Phenomena is shown in Figure 7. This course covers momentum, heat, and mass transport at a theoretical level. The selection of topics and the "resolution" (how detailed the topics are e.g., Energy Flux Vector vs Convective, Conductive and Work Flux Vector) of the topics had to be decided for each course. This was an iterative process to ensure the graphs for each course were similar in terms of the level of resolution. While the graphs made and presented with this approach could look different depending on who makes them, it opens a more formal discussion of what is being taught to students in a course. This could be used alongside learning goals to get the full picture of a course.

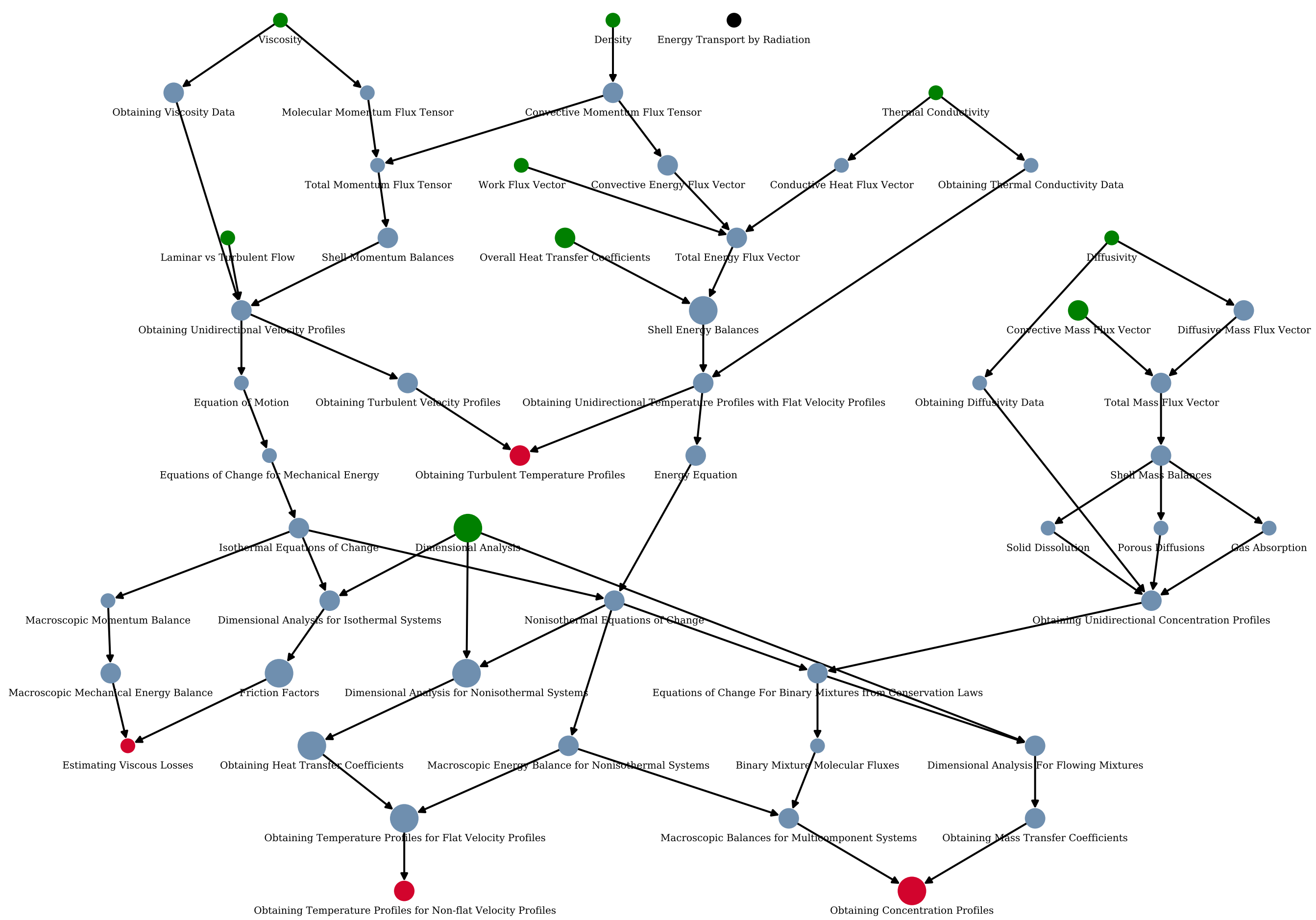

Figure 7: Graph abstraction of CBE 320: Introductory Transport Phenomena.

With a graph abstraction of each course, the topic level graph abstraction of the core chemical engineering courses can be constructed. To do this, the dependencies between courses and review of topics in subsequent courses were identified. For example, there is review of some material from CBE 310: Chemical Process Thermodynamics in CBE 311: Thermodynamics of Mixtures and there are topics introduced in CBE 320: Introductory Transport Phenomena that are required to understand topics in CBE 426: Mass Transfer Operations. This was also an iterative process to ensure topics of the same name had the same scope and to ensure the interconnectivity of courses was fully captured. Again, the interconnectivity of the topics between courses could be up for debate. However, this abstraction encourages conversation beyond "Make sure you introduce the Peng-Robinson Equation of State in Thermodynamics I because I need it for Thermodynamics II" that may occur between instructors teaching a series.

The topic level graph abstraction is presented in Figure 8. Due to the size of the graph, an interactive version of the graph is available in the supplementary material. Across the 13 courses and 40 credit hours, there are 288 topics and 398 edges. 46 topics are introduced in one course and reviewed later in another totaling 4.76 credit hours. Introductory topics often involve a physical property like density and entropy while topics without successors are often lab experimentation or extensions into more complex applications such as multi-phase reactions.

The grouping of topics into modules was conducted restricting the module size to 2.9 to 4.1 credit hours across 13 modules resulting in modules in Figure 9 (unfortunately, due to the size of the problem, the optimization was solved to 12% optimality gap in one hour and future work will need to be done to optimize an entire curriculum with reasonable solver times). The modules here would reflect if the department wanted to keep the same number of courses but rearrange the topics to create more even courses rather than being between one and five credits. Some of the modules are close to what they were before. For example, topics covered within CBE 470 and CBE 450 are largely still with the same topics. This demonstrates these courses are distinct from other parts of the curriculum. For CBE 450, this means that implementing systems thinking and design throughout the curriculum will require a different approach than just rearranging the topics from process design. Further highlighting the differences between process design and systems design.

An additional note is on the transport related courses of CBE 320, CBE 324, CBE 326, CBE 426, and CBE 424. Previous work has discussed how different institutions sequence transport topics. The curriculum is currently designed to introduce the theoretical aspects of transport in one course and equipment and experimentation in subsequent courses. While the modules presented tend to have the theoretical, equipment, and experimentation of each transport type (momentum, heat, mass) together due to the higher connectivity of topics within the same transport type compared to between the transport types.

We have not used the insights from this analysis to impact the current curricula because we are still in the early phases of developing this analysis technique. From these early insights with the transport topics, however, the modules presented reduce the time spent on reviewing the theoretical topics in the equipment and experimentation courses by placing the transport types together. This approach, along with other reorganizations, reduces the time spent on reviewing topics from 4.76 credit hours to 2.24 credit hours for a reduction of 2.52 credit hours. This reduction in credit hours required opens time for

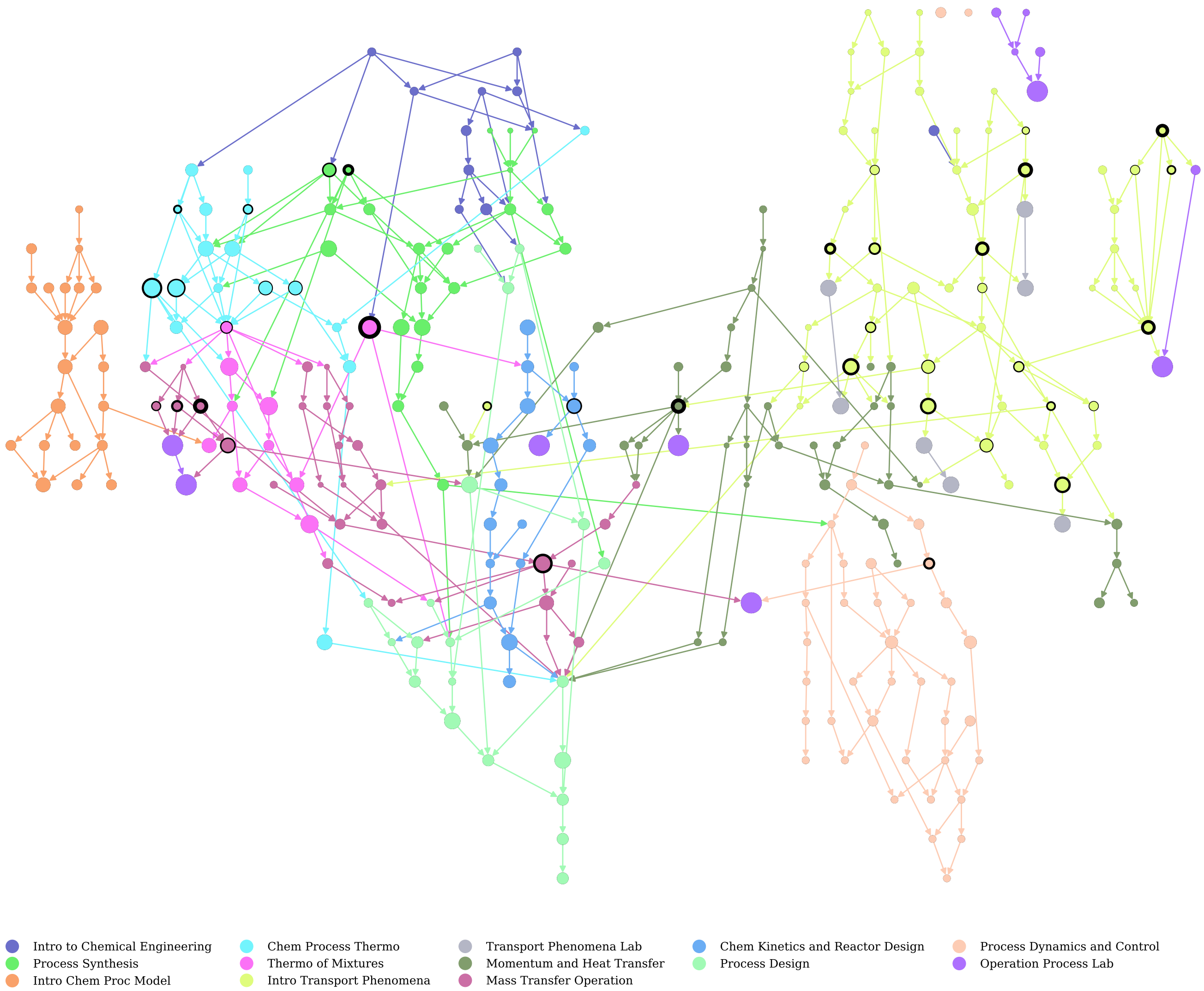

Figure 8: Graph abstraction of the topics covered in the core chemical engineering courses where the topic color represents the course it is currently first introduced in.

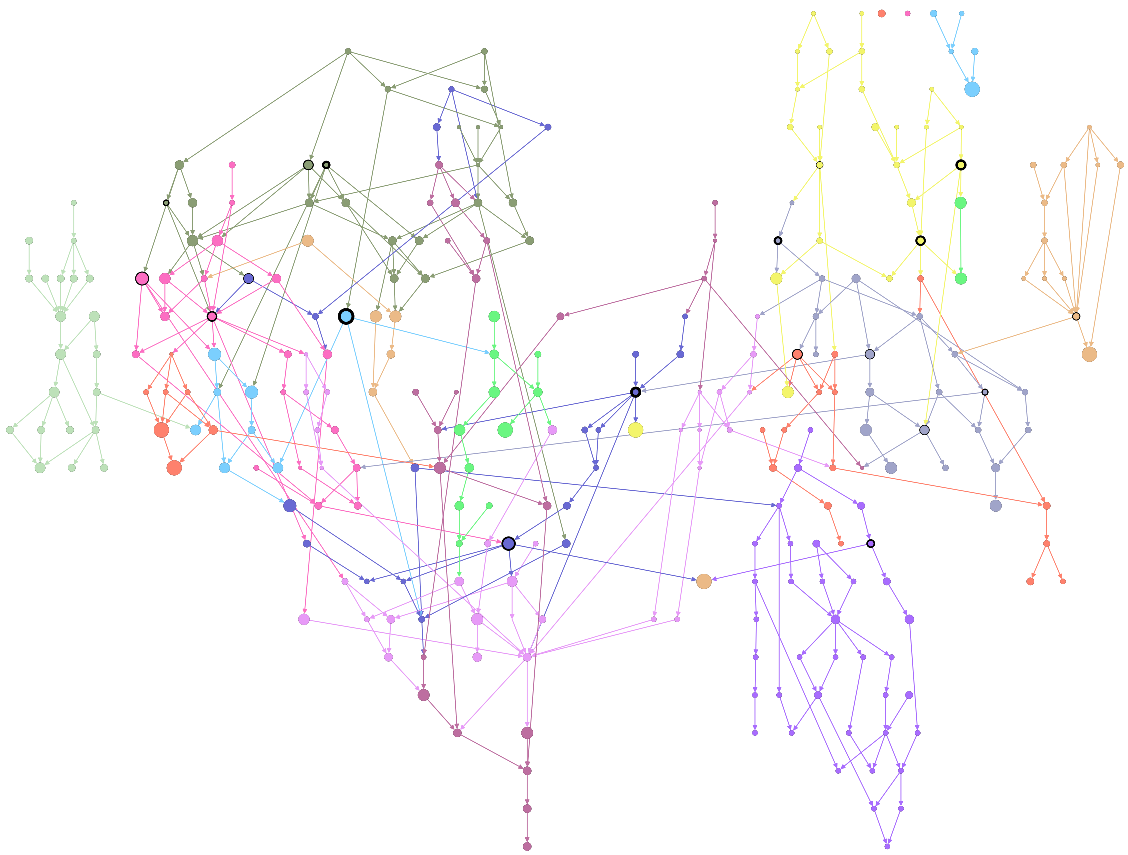

Figure 9: Graph abstraction the core chemical engineering courses with topics grouped into modules

new topics such electrochemistry or systems thinking and design. Implementing these changes would have some difficulties from a logistical aspect with changing courses and student progress and could face resistance from current instructors and potentially accreditation. However, with the formalization of the measures used to evaluate the curricula we can provide some justification and reasoning for the changes.

4 Conclusion

This work presented a graph abstraction in which courses are represented by nodes and connections between courses are represented by edges. Additionally, a graph abstraction in which topics are represented by nodes and topic dependencies are represented by directed edges was presented. This graph abstraction was applied to the core chemical engineering courses required for undergraduate chemical engineering students at a subject university. This abstraction identified lowly and highly connected topics and discussed ways of grouping topics. This abstraction not only encourages instructors to discuss how topics and curricula are presented but also provides a more formal framework and measures to do so.

5 Future Work

While the core chemical engineering courses for the undergraduate chemical engineering program at the subject university was presented here, this analysis can easily and will be expanded to include the entire undergraduate chemical engineering curriculum. This will enable further study into the interconnectivity between chemical engineering courses and background courses from chemistry, math, and physics. With this ability, the determination of which courses should be taught by the chemical engineering department and which courses are appropriate to be taken from other departments.

For example, the chemical engineering undergraduates are required to take the three-credit STAT 324: Introductory Applied Statistics for Engineers course while there also exists CBE 562: Statistics for Chemical Engineers which is a three-credit course which can be used towards the chemical engineering elective requirement. This graph abstraction could be used with connectivity measures to make a case for CBE 562 to replace STAT 324 entirely to improve the connectivity of the topics with more relevant chemical engineering examples and improve reinforcement of topics.

In the current framework, some additional constraints could be added to account for additional attributes of topics. For example, a topic could be categorized as being a lab or lecture topic and a constraint could be implemented to ensure modules are made such that they are either lab or lecture modules. This would make the modules more accurately reflect more traditional course organization.

While the module sizes were set to the sizes of typical courses in this analysis, the module size can be smaller. Setting the module size smaller would reflect smaller units within a course. This would enable writing specific learning objectives for each module and packaging the curriculum into more easily presented sizes. This would further encourage discourse surrounding the curriculum and what exactly is and should be taught within it. The creation of smaller modules could have the added benefit of increased flexibility to move modules around and replace modules with different concepts such as adding an electrochemistry module to the curriculum.

Another use for this graph abstraction is to embed important concepts or ideas throughout it such as safety, sustainability, and systems thinking and design. For example, implanting systems thinking and design throughout the curriculum can be accomplished by creating case studies. These case studies would involve systems thinking and design that depend on topics from different stages in the curriculum. In doing so, the case studies would have to be designed in a way such that all topics are used to understand and solve the case studies. A similar approach could be used for implementing safety and sustainability.

We created the figures in this paper using a set of csv files detailing the topic names, topic sizes, and dependencies for each course. Then the reviewed topics and sizes are determined and a graph is built using the networkx package in python. A combination of the python packages matplotlib, networkx, and pyvis are then used to create the static and interactive visual representations of the graph. We are interested in creating an interactive tool in which someone could build the course graphs by dragging and dropping nodes and connecting the nodes and then use built in tools to determine the layout of the nodes and creating the modules. This would enable other educators to more easily create graph representations of their courses and curricula and determine new ways of grouping the topics.

Supplementary Material

Interactive HTML files of Figures 8 and 9 are available at https://github.com/zavalab/ML/tree/master/CurriculaGraphs.

Acknowledgments

We acknowledge funding from NSF CAREER award CBET-1748516

References

[1] C. Cohen, "The early history of chemical engineering: a reassessment," *The British Journal for the History of Science*, vol. 29, pp. 171–194, June 1996.

[2] I. S. Rampasso, R. Anholon, D. Silva, R. E. Cooper Ordoñez, O. L. G. Quelhas, W. Leal Filho, and L. A. Santa-Eulália, "An analysis of the difficulties associated to sustainability insertion in engineering education: Examples from HEIs in Brazil," *Journal of Cleaner Production*, vol. 193, pp. 363–371, Aug. 2018.

[3] R. C. Armstrong, "A Vision of the Chemical Engineering Curriculum of the Future," *Chemical Engineering Education*, vol. 40, pp. 104–109, Apr. 2006. Number: 2.

[4] V. G. Gomes, G. W. Barton, J. G. Petrie, J. Romagnoli, P. Holt, A. Abbas, B. Cohen, A. T. Harris, B. S. Haynes, T. A. G. Langrish, J. Orellana, H. T. See, M. Valix, and D. White, "Chemical Engineering Curriculum Renewal," *Education for Chemical Engineers*, vol. 1, pp. 116–125, Jan. 2006.

[5] C. Arnaud, "Is it time to leave behind chemical engineering accreditation?," Dec. 2017.

[6] J. M. Rouly, H. Rangwala, and A. Johri, "What Are We Teaching? Automated Evaluation of CS Curricula Content Using Topic Modeling," in *Proceedings of the eleventh annual International Conference on International Computing Education Research*, ICER '15, (New York, NY, USA), pp. 189–197, Association for Computing Machinery, Aug. 2015.

[7] S. L. Toral, M. R. Martínez-Torres, F. Barrero, S. Gallardo, and M. J. Durán, "An electronic engineering curriculum design based on concept-mapping techniques," *International Journal of Technology and Design Education*, vol. 17, pp. 341–356, Sept. 2007.

[8] C. S. d. Blas, D. G. Gonzalez, and R. C. Herrero, "Network analysis: An indispensable tool for curricula design. A real case-study of the degree on mathematics at the URJC in Spain," *PLOS ONE*, vol. 16, p. e0248208, Mar. 2021. Publisher: Public Library of Science.

[9] P. R. Aldrich, "The curriculum prerequisite network: a tool for visualizing and analyzing academic curricula," Aug. 2014. arXiv:1408.5340 [physics].

[10] M. Zouri and A. Ferworn, "An Ontology-Based Approach for Curriculum Mapping in Higher Education," in *2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC)*, pp. 0141–0147, Jan. 2021.

[11] N. Kravchenko, H. Alekseeva, and L. Gorbatyuk, "Curriculum Optimization by the Criteria of Maximizing Professional Value and the Connection Coefficient of Educational Elements, Using Software Tools," in *ICT in Education, Research, and Industrial Applications*, vol. 1, pp. 365–378, May 2018.

[12] M. Bussemaker, N. Trokanas, and F. Cecelja, "An ontological approach to chemical engineering curriculum development," *Computers & Chemical Engineering*, vol. 106, pp. 927–941, Nov. 2017.

[13] S. Muryanto, "Concept Mapping: An Interesting and Useful Learning Tool for Chemical Engineering Laboratories," *International Journal of Engineering Education*, vol. 22, pp. 979–985, May 2006.

[14] R. S. Voronov, S. Basuray, G. Obuskovic, L. Simon, R. B. Barat, and E. Bilgili, "Statistical analysis of undergraduate chemical engineering curricula of United States of America universities: Trends and observations," *Education for Chemical Engineers*, vol. 20, pp. 1–10, July 2017.

[15] Y. Shao and V. M. Zavala, "Modularity measures: Concepts, computation, and applications to manufacturing systems," *AIChE Journal*, vol. 66, no. 6, p. e16965, 2020. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/aic.16965.